



MediaServer
Graphics, Text, PDF and Photo
Production and Archive
Database System

Technical Description

The Software Construction Company
3810 Hamby Road
Alpharetta, Georgia 30004 (USA)
(770) 751 • 8500 (Voice) (770) 772 • 6800 (FAX)

Table of Contents

Page

Overview and Specifications

Overview	5
Introduction	5
System Description.....	6
Summary.....	7
System Design and Specification.....	10
General	10
System Factors.....	11
Typical System Configuration (Windows Server Platform)	12
File Formats.....	12
Administration.....	14
Access and Security.....	16
Communications.....	17
Query Language Overview.....	17
Display Features	18
Data Flow Into and From the System.....	19
Editing	21
Proofing	22
Performance Issues.....	23
Support Issues.....	23

SCC MediaGrid Search Forms

Autonomy Search Form	25
Production Search Form	28
Assignment Search Form.....	29
5-Level Search Form.....	30
'Any Text' Search Form	31

Autonomy Query Language

Elements of Query Expressions	33
Simple and Explicit Syntax	33
Operator Summary.....	34
Evidence Operators	35
Proximity Operators	36
Concept Operators	36
Operator Reference	37
ACCRUE Operator.....	37
AND Operator	37
> (GREATER THAN) Operator	37
>= (GREATER THAN OR EQUAL TO) Operator.....	38
< (LESS THAN) Operator.....	38
<= (LESS THAN OR EQUAL TO) Operator.....	38
IN Operator.....	38
NEAR Operator	39

Table of Contents

	Page
NEAR/n Operator	39
OR Operator	40
Paragraph Operator.....	40
PHRASE Operator.....	40
SENTENCE Operator	41
SOUNDEX Operator.....	41
STEM Operator	41
THESAURUS Operator	42
TYPO/n Operator	42
WILDCARD Operator	42
WORD Operator.....	44
Modifier Reference	44
CASE Modifier.....	44
MANY Modifier.....	44
NOT Modifier.....	45
ORDER Modifier	45
Score Operators.....	45
YESNO Operator.....	46
Product Operator	46
SUM Operator	46
COMPLEMENT Operator	47
Natural Language Operators	47
FREETEXT Operator	47
LIKE Operator.....	48
Precedence Evaluation Rules	48
Precedence Rules.....	48
Parentheses in Expressions.....	48
Prefix and Infix Notation.....	49
Commas in Expressions	49
Delimiters in Expressions	49
Angle Brackets for Operators.....	49
Double Quotes for Reserved Words.....	50
Backslashes for Special Characters.....	50

Overview and Specifications

This document provides a basic description of the features and a discussion of issues surrounding the operation of the SCC MediaServer system, which incorporates the MediaGrid client, the MediaFactory file processing and delivery subsystem, and additional components such as the MediaServer Web Browser Access Module, MediaServer Assignment Module, MediaServer Delivery Module, MediaServer TearSheets Module, MediaServer Replication Module, Microsoft Word Integration Module, MediaServer NewsML Module, MediaServer Administration tools, and the Autonomy Search Engine. MediaServer runs on Microsoft Windows Server using Microsoft SQL Server, or on RedHat Linux using Oracle 10g database.

Overview

This section provides a brief overview of the MediaServer product, identifying some of the benefits it brings to a professional publishing workflow environment.

Introduction

The Software Construction Company (SCC) has created a complete workflow management system based on its MediaServer suite of software. SCC believes that this system is ideally suited to newspaper and magazine publishers for a number of important reasons:

1. The system defines a true workflow, preserving familiar working methods and practices as well as bringing together functionality into a single client/server application that previously required multiple clients and database applications. For example, a user can now browse live incoming pictures and graphics with the same software used to browse incoming text, plus at the same time search for archived media files, create and track assignments, monitor page proofs, send files to web sites or for resale to online vendors, and track and deliver electronic tearsheets. Functions such as previewing, comparing, and soft cropping can now take place on screen and in low-resolution without the need for expensive prints or the need to open the high-resolution file (and lose quality through repeated compression).
2. The use of a database allows multimedia files to be copied or moved “virtually”, without wasteful duplication of data, by setting file status and routing information that either match or oppose a user’s login criteria.
3. Designed around proven industry-standard databases, the system is ideally suited for large numbers of concurrent users spread across a wide area network topology. MediaServer clients connect to the server via a TCP/IP connection alone and do not require access to shared server directories. Integral support of Storage Area Network (SAN), Network Attached Storage (NAS), and CD/DVD JukeBox technology allows for large capacity, expansion, reliability, and scalability.
4. MediaServer includes integral support for clustering technologies as well as an option to provide off-site server replication, providing the necessary levels of data and hardware redundancy required by publishing companies operating in a mission-critical environment.
5. SCC’s development cycle is in line with and highly focused on the future requirements of newspaper and magazine publishers.

System Description

The system consists of multiple components:

MediaServer, which operates on Windows 2000/2003 Server with Microsoft SQL Server or RedHat Linux with Oracle 10g, and incorporates state of the art search and retrieval technology from Autonomy, Inc. (formerly, Verity, Inc.).

MediaFactory, which is a multi-channel batch conversion, database insertion, and file delivery application designed to process text, graphics and photos originally transmitted from wire services such as Reuters, AFP, AP, etc., from third-party editorial systems, from legacy archive systems, from systems exporting NewsML, or saved directly from word processors and scanning software. MediaServer performs much of the server-side processing in the MediaServer system. Thumbnails, previews, and text descriptions are embedded, where possible, in the original file. MediaFactory also supports extraction of text, thumbnails and previews from EPS graphics or adverts, as well as completed pages stored in Acrobat PDF format. Processed files from MediaFactory can be saved in folders or inserted directly into the database.

MediaGrid, the cross-platform client that allows dynamic searching of file folders and database libraries for management of multimedia files. It supports powerful features such as customizable editing screens and print layouts, user definable fields, batch record editing, previewing and comparing, image soft cropping, drag-and-drop linking, etc. MediaGrid also has many advanced features such as operating drop palettes, the creation of AutoFill editing macros and Photoshop actions, and other workflow file delivery automations.

The **MediaServer Web Browser Access Module**, which provides access to the SCC MediaServer system via a standard Web browser. The module supports multiple library searches, preview and compare, record editing, image soft cropping, file download, and delivery as well as many customizable display layout features.

The **MediaServer Delivery Module**, which enables automated delivery of documents stored within an SCC MediaServer system. The module supports multiple destinations, each configured with a time to send, an output format, and an output location, including print queue, network directory, FTP site, email address, or IP socket address. The module also incorporates a sophisticated event management system that allows "trigger" conditions to be set based on date and time or on matching field attribute values within the records themselves. Events such as image rotation, multi-object linking, word count calculation, lead paragraph extraction, header field population, unique ID generation, text and image format transformation, and file delivery by email or FTP can all be configured to run automatically.

The **MediaServer Assignment & Tracking Module**, which allows the creation of an assignment from MediaGrid, a Web browser, an editorial system, an email form, or from any system capable of outputting a structured text document to a server directory. The assignment can be tracked through various stages and allows for the attachment of photos and graphics. All submitted files as well as the assignment are linked together for easy retrieval.

The **MediaServer TearSheets Module**, which supports the insertion of ad data exported from SCS Layout 8000, Managing Editor ALS, or from any front-end advertising system capable of producing a delimited export file. This module provides automatic linking between the ad record and related page, ad and invoice components. Once linked, selected components such as page and invoice can be automatically delivered by FTP (File Transfer Protocol) or email to one or more destinations.

The **MediaServer Replication Module**, which provides the ability to maintain an identical copy of a MediaServer system at a remote site. The module incorporates a scheduling mechanism that can be configured to provide near real-time replication of the system, or configured to run only at certain times of the day.

The **MediaServer Microsoft Word Integration Module**, which provides a bridge that allows Microsoft Word to access, display, and change attributes of objects stored in the SCC MediaServer system.

The **NewsML Module** consists of functionality built into the MediaServer, MediaFactory and the MediaGrid client specifically designed to read and interpret NewsML data to extract content and perform hierarchical page to story to photo and graphic linking. MediaFactory's NewsML Channel provides the interface engine of the MediaServer NewsML Module. It processes export packages from NewsML-equipped editorial systems, such as the CCI Editorial System. MediaGrid organizes and displays page and page element relationships.

The **MediaServer Objects Module**, which is a COM (Component Object Model) library of objects that provides programmer access to all the fundamental features and functions of the MediaServer database system. This module is intended for use only by experienced programmers who need to create completely customized applications and interfaces to the MediaServer database.

The **MediaServer User Manager**, which allows the definition of users by username, password, and role, where roles are similar to user groups. User permissions can be set separately for each MediaServer library.

Summary

SCC MediaServer is a fully integrated Text, Photo, PDF Page, Graphics, Sound, and Video archive, centered around an industry-standard SQL relational database and incorporating free-text search technology licensed from Autonomy, Inc.

Access to the MediaServer database is available from the SCC MediaGrid client (for Windows and Macintosh) or from a standard Web browser.

Files for inclusion in the SCC MediaServer database are simply saved to one or more "hot" network directories from where they are automatically inserted and cataloged within the database. Text and preview metadata is extracted automatically during insertion and indexed in real-time for immediate full text searching. Files from live wire feeds can be directed into one or more of these network directories for automatic insertion into the SCC MediaServer database.

The system incorporates a sophisticated event management system that allows "trigger" conditions to be set based on date and time or matching field attribute values within the records themselves. Events such as image rotation, multi-object linking, word count calculation, lead paragraph extraction, header field population, unique ID generation, text and image format transformation, and file delivery by email or FTP can all be configured to run automatically.

SCC has developed interfaces and successfully migrated existing archive data from the following text and photo archiving systems:

Photo Systems -- AP Preserver, AGT, AXS, Cumulus, Extensis Portfolio, Fetch, MediaBank, MerlinOne, Phrasea, Webnative Venture.

Text Systems -- NewsView Solutions FolioViews, Newsbank SAVE, SII LASR, Data Times, Stauffer Gold.

In addition, MediaServer has been integrated with many third-party editorial and pagination systems. These include:

Managing Editor ALS	Net-Lynx
Agile	Open Pages
Atex MediaCommand	P.Ink
ATS	Quark Publishing System (QPS)
Baseview	SII
CCI	SCS Layout 8000
CompuText	Tera
DTI	Unisys Hermes

The SCC MediaGrid client provides the highest performance interface to the system and is aimed at power users such as Librarians, Photo and Graphics Editors, etc. The application contains many powerful features designed to simplify the data enhancing, file processing, and delivery processes.

MediaGrid is cross-compiled and is essentially identical on Windows 2000/XP and Macintosh OSX operating systems. The Macintosh OSX version of MediaGrid runs native on either a PowerPC or Intel Macintosh. Approximately 95% of the software code is written independently of the operating system leading to availability of updates for all platforms almost simultaneously. Platform-optimized libraries and sophisticated cross-platform compiler technology ensure the best possible performance across all supported platforms.

MediaServer can be used as a Wire Browsing system, a Production system, an Archive system, or all at the same time. It allows the system administrator to create “Libraries” that can be used to logically organize data. Libraries used to monitor incoming wire feeds can be set with an automatic purge process according to the user’s requirements. For example, the system might contain wire text, graphics, and photo libraries for browsing and searching files from agencies such as AP, Reuters, AFP, etc. The system might also contain staff photo and text libraries to store internally produced material either live, archived, or both. The system might include advertising libraries containing completed adverts consisting of component graphics and text, or a completed pages library to store PDF versions of every page published by the newspaper or magazine. Multimedia libraries might contain QuickTime or MPEG videos and WAV or MP3 sound clips of current news events.

The structure of the system and the division of “Libraries” is for management and organizational purposes only. Access privileges to these libraries are set by the system administrator using the MediaServer User Manager. In addition, the administrator can further customize each library by defining any number of additional field attributes made up of text strings, text lists, date/number fields, and so on.

In the case of JPEG images received from wire services, digital cameras, or scanned by photographers, these typically contain embedded metadata information commonly known as the IPTC caption. This information is displayed within Adobe Photoshop’s File Info command as well as within SCC’s Plugin Kit for Photoshop. It is extracted automatically during insertion into the MediaServer database and used to populate the image record header. Modifications made to the record header after insertion, whether directly by a user or by an automated server-side process designed to update headers, are automatically reinserted into the image file during export, delivery, or launch in order to update the embedded IPTC information. This ensures that the caption contained in the downloaded file always reflects the latest updates to the record header.

MediaServer also supports the extraction of embedded metadata from many other commonly used file types, such as Photoshop-generated raster images, EPS vector graphics, Acrobat PDF files, QuickTime movies, plus IPTC, XML, tagged, delimited, and plain text formats. MediaServer supports the insertion of files without embedded metadata, in which case the file is inserted with a descriptive icon as its thumbnail and is searchable by filename and insertion time as well as any manually or automatically entered data updates, such as updates by ServerOps or the MediaFactory Updater channel.

The Autonomy search engine, licensed from Autonomy, Inc. and employed within MediaServer, allows full text searching on every record contained in the database. Files added into the system are automatically indexed and immediately available for searching. A variety of sophisticated search capabilities are available including word, synonym, root word, wildcard, proximity, phonetic (fuzzy logic), natural language, boolean, field, and date range searches. Powerful technology unique to MediaServer allows any Search to be set in dynamic mode. This means that even if a match is not found during the first search, when files matching a user's search criteria arrive in the database library the results are delivered automatically to that user's client. Search criteria can be stored and then recalled quickly from disk, restored from a history of the previous queries, or from a memory pipeline by using MediaGrid's forward and back search history buttons on the query result screen.

Alternatively, the user can invoke the Production search form to, over time, continually retrieve objects that have been inserted within any range of 1 to 672 hours (4 weeks). Enabling dynamic mode in the production search again causes the result screen to automatically update as new objects are inserted into the library.

Both MediaGrid and MediaFactory support a programmable "AutoFill" environment that allows the creation of scripted attribute updates for use during the file insertion and the data enhancing process. AutoFill instructions greatly reduce the amount of editing time and increase the accuracy of repetitive tasks such as word count calculation, lead paragraph extraction, adding pre-determined values to fields such as Byline, Title, Date and Time, and changing record status. Selecting one or more records from the search results screen and then picking the relevant item from the AutoFill menu in MediaGrid causes the batch update to take place. The same AutoFill functionality can be automated within MediaFactory.

MediaGrid includes a task automation command that offers a set of functions that can be linked together to create a custom command. For example, the automation command can be configured to bring up an attribute dialog to elicit user input, run an AutoFill command, print a proof, export a customized delimited file containing all the attributes, and then copy the file into a folder that is monitored by a third-party OPI/Pagination system.

The Launch Action feature of the MediaGrid client creates a powerful interface to Adobe Photoshop. Using the Actions feature of Photoshop 5.0 or later, MediaGrid effectively applies any set of Photoshop commands to one or more selected images. This can be used to resample, filter, color balance, or convert to another image file format or color space. In addition, Launch Action is integrated with the SCC Plug-In Kit for Photoshop enabling automatic cropping of the high-res image according to the soft crop defined in MediaGrid or the Web Browser Access module.

The MediaGrid Preview Window displays large images for viewing and managing the advanced cropping features. Images can be temporarily rotated for "right side up" cropping of incorrectly oriented photos. When using the Preview Window, the cropped part of an image can be copied onto the clipboard or into a low-resolution JPEG file. Images can be optionally dimmed or blackened outside of the cropped region for better visualization. Preset crop definitions can be created and recalled at any time to quickly apply commonly used crop dimensions.

The MediaGrid Compare Window allows a user to quickly choose the best files by scrolling through a selected subset of search results and "pinning down" the most suitable images. The Compare Window allows "side by side" comparison, "top to bottom" comparison, or "two by two" comparison of the image or its text attributes. When the best files have been "pinned" in the Compare Window they are subsequently selected within the Grid Window from where they can be cropped, published, printed, copied, deleted, and so on.

Both the Preview and Compare windows allow images to be previewed in either color or black and white.

The user can layout custom attribute dialogs by using a separate designer application (included with MediaGrid). Several attribute dialogs are shipped as standard and may be customized as desired. User defined attributes created within MediaServer are immediately available for incorporation into attribute dialogs.

Users can define and configure popup lists that pull down from any attribute in an attribute dialog. For example, a frequently used list of categories can be presented in a pull down from the Category field within the editing dialog.

Dependencies can also be defined that make the contents of a popup list depend on the value entered in another field. For example, the list of available subcategories can be made to depend on the contents entered into the Category field. Using Popup Lists reduces typing time and increases the accuracy of attribute entry.

Attribute Display Clusters efficiently display a collection of field attributes and user defined text. For example, related attributes, such as filename, file size, and modification date can be packed together into one neatly formatted cell. An attribute display cluster can then be used to view a record's attributes in Multi-Column viewing mode.

Extended Cell Markers allow the creation of colored corner flags that optionally appear over thumbnails whenever certain matching criteria is met. A typical use might be to identify and mark images that are protected, published, copyrighted, or edited.

The Derived Search function allows the creation of saved Autonomy searches that are derived from and depend on the attribute contents of a selected record. For example, a derived search on "byline" could extract the byline content of a selected record and use it to find all other records containing the same byline value. Or, a derived search could be setup to extract the last name of the Byline and search for all records that contain the same last name in the Byline, or in other fields.

The user can layout custom printed page formats by using a separate designer application (included with MediaGrid). A single proof can contain data from several files to create catalog pages or contact sheets or just a single, selected record. Several print formats are shipped as standard that may be customized as necessary.

An extensive set of user options and preferences are supported, making it possible to configure and customize the look, behavior, and operation of MediaGrid. Preferences are stored and can be password protected to prevent changes to a critical configuration setup. This is useful to supervisors that want to maintain consistent preferences over an entire network of workstations.

System Design and Specification

This section provides a discussion of features, file support, performance issues, and operational issues as they pertain to the MediaServer system.

General

MediaServer has been designed to be self-maintaining and requires minimum administration other than normal monitoring and backup procedures. When migrating from older systems, the user will see a smooth transition and a substantial increase in productivity. Legacy and other existing files and metadata are batch processed by MediaFactory and automatically inserted into the MediaServer database.

SCC uses sophisticated cross-platform software development technology to ensure that MediaServer client and server applications and components are virtually identical, where possible and appropriate, and rapidly available across all supported operating systems and environments.

The SCC MediaServer Web Browser Access Module is an optional ISAPI extension for Microsoft Internet Information Server, which provides access to the database via industry standard Web browsers such as Netscape Navigator, Safari and Microsoft Internet Explorer.

System Factors

The MediaServer system requires at least one dual or quad Intel processor server (either Windows 2000/2003 or RedHat Linux) and will benefit greatly from having a large amount of RAM - 2GB minimum (more is better) and requires sufficient disk storage to accommodate the SQL database, search index, and high-resolution files.

For the purpose of calculating disk space requirements, there are essentially two storage areas: one for the database and one for the high-resolution file system (typically both RAID 5 devices). These storage areas can reside on the same server or optionally be located on separate standalone servers (clustered together if failover is required).

The first storage area is referred to as the "SCCDB" device and is used to hold the SQL databases, transaction logs, and the Autonomy subsystem. The storage requirements for this device can be estimated based on 100KB for each record that contains previews (i.e. photo, graphic, page, ad, etc.) and 15Kb for each text record. In calculating this figure, any existing image and text data should be included, as well as expected throughput for a suitable number of future years.

The second storage area is referred to as the "SCCData" device and is used to hold the high-resolution version of each stored file (**NOTE:** This device does not store text records, which are stored ONLY in the database). You will need to look at your existing images, graphics, and PDFs to calculate this storage requirement. Note that high-resolution files stored on this device can be migrated later to other network accessible devices such as stand-alone file servers, network attached storage (NAS) devices, or CD/DVD Jukeboxes.

The hardware should include or be accessible by a server having access to a high-performance DLT backup device.

SCC MediaServer is compatible with Microsoft Cluster Server (MSCS). Many MediaServer installations use this technology. While combining multiple server functions on a single machine is possible and commonly done, a clustered configuration provides maximum availability of server resources as well as maximum fault tolerance in data and hardware redundancy by distributing server functions across multiple server nodes. In the event of hardware failure of either node, then the cluster will move all services from the failed node to the working node. In an SCC Cluster configuration under normal operating conditions, one of the nodes in the cluster acts as the Database Server while the other acts as the Data Server. In the event of hardware failure of either node, the working node will take over both roles automatically.

In addition to hardware, the following third-party software is required (for Windows Server platform):

- Microsoft Windows 2000/2003 Server plus client access licenses (**Note:** Windows 2000/2003 Advanced Server is needed if clustering is required. Advanced Server is also required for servers with memory configurations over 4GB RAM)
- Microsoft SQL Server Standard Edition plus client access licenses or processor licenses. (**Note:** SQL Server Enterprise Edition is needed if clustering is required. Enterprise Edition is also required for memory configurations over 2GB RAM)
- Adobe Acrobat or other suitable PDF creating software to extract text and previews from PDF and EPS files
- Veritas BackupEXEC with the SQL Server module (supports online backup of SQL database(s), Autonomy sub-system, and high-resolution files).
- Symantec pcAnywhere32 (required for remote access by SCC technical support personnel).

Typical System Configuration (Windows Server Platform)

MediaServer's basic configuration consists of a Database Server and a Data Server. These two conceptual servers can run on separate server hardware (two machines) or both on the same server hardware (one machine).

Database Server: This server is configured with Windows 2000/2003 Server, Microsoft SQL Server and the SCC MediaServer system, including full text search technology licensed from Autonomy, Inc. Web services are provided via Internet Information Server (IIS), installed during the installation of Windows 2000/2003. SCC's MediaServer ISAPI extension for IIS allows access to the database for searching, browsing, and optional downloading via industry-standard Internet Web browsers such as Netscape Navigator, Safari, and Microsoft Internet Explorer. The SQL database, MediaServer, WWW, and Autonomy components are normally stored on a hot swappable RAID device. Only descriptions, thumbnails, and previews plus supporting data are resident within the database. High-resolution files are stored externally on the Data Server. MediaGrid and Web browser clients are provided access to the high-resolution files for download via multithreaded channels directly through the database. Clients do not need to mount any server volumes to download files.

Data Server: This server is configured with Windows 2000/2003 Server, IIS FTP services, and SCC MediaFactory. FTP and MediaFactory configuration information and all high-resolution data are stored on a hot swappable RAID device.

Additional Storage: It is possible to migrate high-resolution files over time from the Data Server to alternate locations such as Network Attached Storage (NAS) or nearline CD/DVD Jukebox devices.

File Formats

SCC recommends a policy of standardizing file formats where possible before inserting into the MediaServer database and, in the case of JPEG images, embedding attributes such as IPTC caption, text descriptions, and previews inside the original file. This ensures that each file's descriptive information is stored on the chosen long term storage media (e.g. CD/DVD) as well as in the MediaServer database, providing an increased fault tolerance as well as allowing for portability and easy transfer of this media (i.e. CD/DVD) into other database systems.

Photos

In order to optimize storage requirements and provide the fastest possible network access, SCC recommends that photos are stored in a JPEG format containing thumbnail, preview, IPTC caption (plus optional custom attributes), and the compressed high-resolution data in a single file. In most cases the stored file should be the photographers uncropped RGB original (see explanation below). Image files in most other formats can be converted to JPEG using SCC's MediaFactory batch processor. Each MediaFactory channel supports the creation of "AutoFill" definitions allowing default IPTC caption attributes and unique reference numbers to be written to images based on the incoming drop folder.

Quality. There are a number of production related issues, more practical than technical and often open to debate, influencing the storage of images. It cannot be certain in the case of an image in the archive that it will be used for the same purpose each time it is retrieved. For example, an image might be printed in a magazine for its first use, in a newspaper for its second use, and finally in an on-screen presentation. It is consequently felt that in order to capitalize on the quality and reusability of images stored within a database environment they should be saved as uncropped raw RGB images that have had no print dependent dot gain compensation or color separation tables applied. The fact that these "output" processes are so very different requires that an image be prepared differently and so it is advantageous to archive the raw RGB original.

Similarly, the effect of JPEG compression on the quality of an image is also a much discussed topic. While it is true that JPEG is a lossy algorithm, most images originate or are received in a JPEG format either from digital cameras or from wire services. It is important that these images are not recompressed during insertion into the MediaServer database.

Soft Cropping. It is recommended that the original image is stored uncropped, since it will not always be the case that the same portion of the image will be desired for every use. Both MediaGrid and Web browser clients provide a soft crop interface that allows an image to be marked with a non-destructive crop overlay. At the point the image is “published”, indicating that the image is to be used in a given publication, then a copy of this image is forwarded to a production folder for the attention of an operator working in Adobe Photoshop (or similar). At the point the image is opened, the “soft crop” is implemented and the image is sub-sampled, ready for tone correction and color separation. Alternatively, the published file can be post-processed by a MediaFactory image conversion channel configured to read the soft crop information and actually crop the file.

Workflow Automations. Publishing companies are generally interested in streamlining and simplifying workflow where possible. With this in mind, SCC has put a large emphasis on the development of workflow automation tools. These automations vary and include simple folder to folder transfers, applying predetermined “AutoFill” attribute values, triggering file delivery via FTP or email, launching Adobe Photoshop Actions, interacting with Adobe Graphics Server, or any combination of these and many others.

Digital Camera Files

SCC MediaFactory supports the conversion of raw files saved either in the Canon or Kodak DCS digital camera format into either JPEG or TIFF formats for subsequent insertion into the MediaServer database. Alternatively, they can be inserted in their original raw format. In either case, embedded IPTC Caption information is automatically extracted and mapped to database fields during insertion into the MediaServer system. Images from Nikon Professional Series cameras are supported in JPEG format.

Text

SCC MediaFactory supports the normalization and subsequent insertion of text files from wire services, editorial systems, legacy text archives, or any other sources capable of outputting a structured or plain text format. Supported text formats include AP NewsDesk, ANPA Wire Service (Bulletin 1312), Lexis Nexis Feed, text from SII, Atex and Harris editorial systems, NewsView Solutions FolioViews, NewsBank SAVE and DataTimes archive formats, plus standard text formats such as XML, Delimited Text (single record per file), Delimited Text (multiple records per file), and Bounded Text format. The Delimited Text format is a standard used by many applications, such as FileMaker, Microsoft Access, Excel, and most applications that export tabular data. Bounded Text format is a variation of delimited text where each field is bounded by a unique start and end string.

SCC MediaFactory inserter channels include a “Character Mapper” function that allows a user to define custom character translations during file processing, a “Value Translator” function that allows translation and expansion of matching words contained in preconfigured attribute fields, and an “AutoFill” function that allows automated entry and enhancement of data within record attribute fields. In addition to built-in text format support, custom text filters can be developed for unsupported formats.

Pages

SCC MediaFactory supports the insertion of completed pages in Adobe Acrobat PDF format. During the insertion process, thumbnail and preview information together with all text contained in the page is extracted from the file and used to populate the record in the MediaServer database. Text contained in the record is indexed automatically by the Autonomy search engine. Saving pages in PDF format allows anyone on the system to view or print a page using the freely distributed Acrobat Reader without the need for the software application originally used to create it (e.g. QuarkXpress, InDesign, etc.).

MultiFile

“MultiFile” is a special insertion mode available in SCC MediaServer for users who need to access the original document for re-editing in the original package of related files. In this mode, both the original document and any number of other related documents are placed together in a folder (or nest of folders). When SCC MediaFactory recognizes a MultiFile folder within its input directory it searches for a pre-configured “master” file type (e.g. JPEG, EPS, or PDF) and uses this “master” file to create the database record. During the insertion process, thumbnail, preview, and all embedded text is extracted from the “master” file for use in populating the record and for indexing by the Autonomy search engine. All other files and folders contained within the original folder are also inserted and linked to this “master” record. When a MultiFile record is downloaded from either MediaGrid or the Web browser client, a dialog appears that allows the user to select any individual file, or the entire “MultiFile” folder.

Graphics

Vector graphics from applications such as Illustrator and Freehand can be saved in native, editable EPS or PDF formats. They can also be saved as part of a “MultiFile” folder (see paragraph above). In the case of native Illustrator, and editable EPS or PDF files, SCC MediaFactory can automatically extract thumbnail and preview information together with all text contained in the graphic to create the database record. All extracted text is automatically indexed and searchable.

Multimedia

SCC MediaFactory supports automatic extraction of thumbnail previews from QuickTime Movies during insertion into the MediaServer database. All other multimedia formats not containing embedded preview information, such as WAV, MPEG, MP3, WMV, etc., are inserted with a generic thumbnail and searchable by Filename as well as any manually entered information.

Foreign Files

Any other file formats not afforded special support by the system can be inserted into MediaServer with a generic thumbnail, searchable by filename or any manually entered data and later exported or launched.

Administration

Since SCC MediaServer is an open system based on industry-standard operating systems and database technologies, storage capacity can be expanded easily. Additional disk arrays, file servers, Network Attached Storage (NAS), and CD/DVD JukeBox devices can be easily added to the system at any time.

Regarding the configuration of the MediaServer databases and libraries, while there are no restrictions on the size of a library, it is good practice to organize data into multiple libraries to allow for more logical usage. For example, for a system designed to store pages, text articles, photos, and graphics, it is a good idea to create individual libraries for each data type, each having its own field structure and user access permissions. Both MediaGrid and Web browser clients support the ability to search across libraries so, if required, library boundaries can be transparent to the user. This design has the advantage of allowing the system administrator to subdivide the database into file groups, each located on its own disk array and containing one or more libraries. For example, more commonly accessed “live” libraries could be located in a file group on a faster disk array, while less commonly accessed “archive” libraries are located in a file group on another disk array. Veritas BackupEXEC and SQL Server’s own backup functions allow file groups to be backed up and restored independently of each other providing a more granular approach to system backup and recovery. The flexible design of MediaServer even allows data storage to be spread across multiple database servers since the SCC MediaGrid client allows connection to libraries across servers just as seamlessly as attaching to multiple libraries on the same server.

MediaServer is packaged with several complimentary applications. These applications include:

The **SCC MediaServer Administration Utility** is a MediaServer-specific utility for performing administration functions particularly associated with the operation of MediaServer. With this utility the system administrator can create an SQL database, create, rename and delete MediaServer libraries, copy or move data between libraries, optimize Autonomy collections, optimize SQL indexes, and purge “deleted” records all from a simple graphical user-interface. The MediaServer Administration Utility is also used as a diagnostics tool to allow the user to make checks on record counts and search indexes.

The **MediaServer User Manager** allows the creation of logon credentials for individual users or groups of users. It also can be used to assign rights and privileges to users and groups, such as the privilege to view, search, update, delete, create, deliver, and download. Separate privileges are assigned to each library for each user.

The **MediaServer MediaMaker Utility** aids in the management and creation of offline, nearline, and auxiliary storage. High-resolution files are selected automatically by age and other criteria and are relocated to other file servers, Network Attached Storage (NAS), and CD/DVD Jukebox devices. All database pointers and flags are updated to reflect the new storage location.

The **MediaFactory Maintenance channel** performs general day-to-day administration automatically. This channel drastically simplifies MediaServer maintenance and optimization setup, execution, and monitoring. Library maintenance functions include deleting the oldest, unprotected records from a library, purging deleted records, optimizing both SQL table indexes and Autonomy search indexes. The Maintenance Channel can be configured to start execution at any time, usually during the early morning hours, and to stop prior to completion if the functions cannot be completed before a specific time. Any error or completion status that results from function execution is passed back to MediaFactory and displayed in the Log (Monitor) Window as an event for the Maintenance Channel. Error events can be optionally emailed to a recipient list (see below).

SCC MediaFactory has the ability to write a detailed log in addition to maintaining its own application log. This detailed log is written to a separate table within the SQL database and is accessible by a standard Web browser. It displays the log events associated with each single channel or with all channels combined. Each event can be displayed in collapsed format showing the profile of the processing event, or in full expanded format with a complete list of details associated with the processing event.

MediaFactory can be configured to email everyone in a recipient list in the event of an error or a warning, or if it is stopped or started. It can also be configured to email the recipient list should one of several events occur, such as File Could Not Be Deleted, Attributes Could Not Be Retrieved From The File, Input File Is An Unselected Format, or others. Email notification requires that MediaFactory be running on a server or workstation that has been configured as a MAPI-compliant email client.

The SCC MediaServer system uses SQL as its underlying database structure and database management system. Access to data held within is therefore accessible via third-party systems supporting SQL, such as accounting and report generating packages.

SQL Database Management System vendors provide database consistency check tools, such as DBCC, which detect, report, and often repair problems within in the database. This utility can be scheduled for automatic execution on a timed basis using the MediaFactory Maintenance channel.

SCC MediaServer supports several advanced features for the automation of repetitive tasks. Programmable pop-up lists allow easy entry of standard information relevant to particular fields (attributes). Pop-up lists can be further enhanced to allow for the contents of one list to be dependent on the values selected from another. For example, if the ‘category’ attribute displays **Sport** then the ‘keyword list’ could display *football, golf*, etc. However, if the ‘category’ attribute displays **News** then ‘keyword list’ could show *world, local, politics*, etc. An Auto-Complete feature can be turned on so that the values are automatically filled in as the field characters are entered, based on matching words in the attributes pop-up list.

The system also incorporates a store and load default feature that allows the contents of a record to be stored and recalled for reuse at a later time. This allows the contents of most fields, such as Byline, Title, Desk, etc., to be configured and stored in advance, and recalled each time a new caption is created.

Both MediaGrid and MediaFactory applications support a programmable “AutoFill” environment that allows the creation of any number of autofill commands for use during the file insertion and data enhancing process. AutoFill commands greatly reduce the amount of editing time and increases the accuracy of repetitive tasks such as word count calculation, lead paragraph extraction, adding pre-determined values to fields such as Byline, Title, Date and Time, and changing record status. Selecting one or more records from the search results screen and then picking the relevant command from the AutoFill menu in MediaGrid causes the batch update to take place.

Both MediaGrid and Web browser clients support customizable automation routines that can be configured to act on any file or group of files. This routine could involve presenting an edit dialog to the user, performing an AutoFill command, then copying the selected file(s) to another area (e.g., a toning department in the case of images) as well as automatically copying or moving the file to an archive library.

Custom corner flags can be defined to appear as markers on the thumbnails in MediaGrid whenever matching criteria is met. This feature could be used to show certain colored corner flags whenever images are protected, published, copyrighted, or edited, as an example.

SCC applications, including MediaServer, MediaServer Modules, MediaGrid, and MediaFactory, have been developed with state-of-the-art software development tools and techniques. These software applications and components have been designed, implemented, and tested to conform to or exceed industry standards. They are capable of correctly managing, interpreting, manipulating, processing, and storing all date-related data, including data referencing dates prior to, within, or beyond the Year 2000 up to and including 2039, and data involving any combination of such dates. They are capable of correctly managing, interpreting, manipulating, processing and storing, date-related data for the aforementioned dates without resulting in or causing logical or mathematical errors or inconsistencies. Moreover, all date-related data interface functions can be set to include an indication of the century (as a 4-digit year).

Access and Security

Client access to the MediaServer system is supported by either SCC MediaGrid for Windows and Macintosh, or from a standard Web browser (requires MediaServer Web Browser Access Module). The maximum number of users is managed by the number of purchased “Power” and “Lightweight” concurrent user licenses, not by the number of installed client applications. Generally speaking, lightweight users are able to perform searches, view results, issue delivery commands and optionally download files while power users can also crop images, edit attributes using custom editing forms and delete records depending on the individual users login permissions.

MediaServer User Manager allows users to be defined and library-specific permissions to be assigned.

The MediaServer system fully controls access via user login and password. Documents contained within the system can be set as available or unavailable to certain users.

A user’s login and password determines their profile within the MediaServer system, which can either be set per user or inherited as part of group privileges. In addition, a user can be specified as belonging to a certain department, e.g. Sport. From that user’s login they can be limited to searching only on documents that have a category equal to “S”.

System users, groups and passwords are all user defined and changeable. System managers have the ability to reset the password for users.

Each user is assigned access privileges for each individual library, including privileges pertaining to viewing, changing, and deleting.

Communications

The TCP/IP network protocol is used throughout the SCC MediaServer system.

The MediaServer system can be accessed remotely via dial-up using Microsoft Remote Access server (or other), through a VPN, or directly via a Wide Area Network such as leased line or ISDN bridges. Connection is seamless and the medium type is transparent to the client.

Access to the MediaServer system, whether from a web browser or MediaGrid client, is licensed by concurrent user count. When either client logs into the system and issues a search, a connection is reserved on the server. The connection is released when the MediaGrid client logs out, or in the case of the Web browser client, when there has been no activity for a period of time. Therefore, a 10 user concurrent connection license would enable up to 10 users to search simultaneously, regardless of whether they are using MediaGrid or Web browser clients. Based on the number purchased, servers are configured with a maximum allowable number of "Power" Users and a maximum allowable number of "Lightweight" Users. MediaGrid and Web browser clients support both power and lightweight user connections. Generally speaking, lightweight users are able to perform searches, view results, issue delivery commands, and optionally download files while power users are additionally able to crop images, edit attributes using custom editing forms, and delete records.

Query Language Overview

SCC MediaServer incorporates the industry-leading Autonomy search engine, which conforms to widely accepted standards for structured text searching. Searches can be of a simple format in which the query is a phrase, producing results that contain only that phrase and no irrelevant hits. Searches can also be of a more complex nature containing advanced query components such as boolean values, natural language, wildcards, word stemming, phonetic (fuzzy) searching, synonym, proximity, and so on.

The system supports the following types of searches:

Literal phrase or "string" searches.

Natural language searches. In this mode, every word within the search is parsed out and treated independently by the query engine. Those documents containing the largest number of these words and occur in higher frequency and density are considered to be more relevant and therefore are given a higher score than other documents. Result sets sorted on descending score show the most relevant documents at the top and the least relevant at the bottom (Natural language searches are similar to those performed by most Web search engines).

Single or multi-character wild cards. Use the '?' and '*' wildcards.

Word stemming. Also called "root word expansion", the <STEM> operation is applied by default to each word. A search for 'ends' will find a match on 'end', 'ended', 'ending', but not 'endeavor', 'endear,' etc. Word stemming can be enabled or disabled as required.

Adjacency or proximity searches. Issues proximity searches via the <NEAR / x> command where x is the number of words between the two words of the query.

Lyndon <NEAR / 2> Johnson

searches for Lyndon Johnson, Lyndon B. Johnson, and Lyndon Baines Johnson. When the <NEAR / x> command is not specified the system defaults to simple phrase searching.

Boolean arguments. Use boolean search logic, such as AND, OR, NOT, <, <=, >, >=, etc. The boolean search can also be extended through the use of parentheses to indicate the precedence of the search operators. Information within parentheses is read first. The following query example means: *look for documents that contain a and b, or documents that contain c.*

(a AND b) OR c

Nested parentheses: The following query example means: *look for documents that contain b or c as well as a, or that contain d.*

(a AND (b OR c)) OR d

Field searches. The system also supports further refining of a search via the <IN> command by matching the specified word or phrase with values stored in a specific document field. For example,

(Dallas Morning News) <IN> Publication

The “LIKE” search feature: A powerful search feature that allows a user to quickly and easily select an object and query for all objects that are “similar to” the selected object.

Date range searches: Constrain results with date ranges, including content date, insertion date, file creation date, last publication date, last modified date, or any custom date field created by the system administrator. Special operators and expressions are recognized, such as ‘today’, ‘tomorrow’, and ‘yesterday’, or ‘today - 90’ (meaning 90 days prior), to specify date ranges and other query phrases.

Search criteria can be stored and then recalled quickly from disk, restored from a history of the previous 25 queries, or from a memory pipeline by using MediaGrid’s forward and back search history buttons.

A detailed description of the Autonomy Query Language is provided in a later section of this document.

Display Features

Both MediaGrid and Web browser clients offer flexible sorting of search results, such as by creation date, insertion date, content date, score, or any short text attribute, ordered as ascending or descending, and mixing multiple library results. Any of these sort orders can be stored as default or applied on command.

Both MediaGrid and Web browser clients supports various display options including:

- **Thumbnail grid view**, which can display with or without the filename.
- **Single or multiple line multi-column list view**, which displays a user-definable selection of attribute fields (spreadsheet view) with or without the thumbnail.
- **Summary view**, which shows a list of sentences that contain words and phrases matching the query
- **Text and image preview modes**, which allow the user to switch between a large image view and a user-definable selection of attribute fields. Graphic objects, such as photos, graphics, and pages, default to image preview mode while text records default to text preview mode.
- **Compare mode**, which displays a side-by-side large format display of multiple selected records for comparison purposes. As with the preview mode, the user can switch between a large image preview or a user-definable selection of attribute fields in this mode.

NOTE: Many variations of the above displays can be created via MediaGrid’s integral “Attribute Cluster Editor” and using the Web Browser Module’s compound attribute definition controls. The administrator can create groups of attribute fields for either client in a variation of fonts, sizes, and colors. Records can also be displayed and/or edited in user-definable dialogs as well as printed to Postscript printers on user-designed page formats containing all or a subset of attributes, images, variables, and static text.

An optional Info Bar can be displayed at the top of each MediaGrid results window. The Info Bar provides one more screen area where the user can control the display of information. This display area can be dragged to any vertical size and can display any single attribute of the selected cell, or any combination of attributes.

Words and phrases matching the search criteria are automatically highlighted in multi-column list view, preview and compare modes, as well as MediaGrid’s Info Bar.

Data Flow Into and From the System

The MediaServer system supports the export of single records or multiple records in batches, either client-side via the execution of a single command, or server-side using the MediaServer Delivery Module.

SCC MediaServer supports the printing of single records or multiple records in batches, either client-side via the execution of a single command, or server-side using the MediaServer Delivery Module.

SCC MediaServer supports delivery by email or FTP of single records or multiple records in batches, either client-side via the execution of a single command, or server-side using the MediaServer Delivery Module.

The Delivery Module can be invoked manually with a single command or automatically with a persistent command to export, print, email, or FTP records or objects. Any number of persistent commands can be configured to trigger automatically based on time and/or matching criteria within records.

SCC MediaFactory can be configured to poll network directories for files to be inserted into the system. Images, vector graphics (either from wire services or produced internally), and completed pages saved in Acrobat PDF format are inserted with embedded metadata fields automatically mapped to attribute fields in the database. Thumbnails and previews are generated and all extracted text is automatically indexed and searchable with the full power of the Autonomy search engine used by MediaServer.

SCC MediaFactory's Text Conversion channel can process text from wire services, editorial systems, or legacy text archive systems, either in single or as batch documents. Character and text string translations can be performed automatically using "Character Mapper" functions that allow a user to define custom character translations, "Value Translator" functions that perform translation and expansion of matching words contained in preconfigured attribute fields, and "AutoFill" functions that allow automated entry and enhancement of data within record attribute fields. Text files are parsed and data contained within fields is automatically mapped to attribute fields in the database. Custom translations not already supported by MediaFactory's existing text conversion filters can be handled with the development of custom filters. Depending on their complexity, some custom filters will incur an additional fee.

A MultiFile is a single object in the database that is actually composed of several files and possibly several folders of files. It can be inserted by MediaFactory into any MediaServer library. In both MediaGrid and Web browser clients, the MultiFile is viewed as a single object, represented by a single thumbnail, and associated with one set of text attributes. When a MultiFile object is selected for download from either client, a special dialog is presented showing a tree structure of the MultiFile folder of files and subfolders. From this dialog, you can download the entire MultiFile, resulting in the entire folder being copied to your destination path, or you can download a single selected file from inside the MultiFile folder structure. With MultiFile support you can archive entire folders of related files as a single object in the database.

SCC MediaFactory's MultiFile insertion mode recognizes folders as MultiFile documents. When a MultiFile folder is detected, the MediaFactory inserter channel will look for a pre-configured "master" file type (e.g. JPEG, EPS, or PDF) and use this "master" file to create the database record. Thumbnails, previews and text are extracted from the master file for use in the indexing of the record. All other files and folders contained within the MultiFile folder are linked to this "master" record.

The SCC MediaServer system accommodates photos, graphics, text, pages, sound, and video in the following formats:

Photos: RGB, CMYK, Grayscale and Single Bit TIFF (uncompressed), JPEG, dual-file JPEG with external attributes (as supplied by the Reuters RMR service), Photoshop EPS and EPS JPEG, AP LeafDesk, IPTC, Canon and Kodak DCS digital camera format, and GIF, plus most other image formats saved from Adobe Photoshop for Macintosh. Embedded IPTC Caption fields map automatically to the appropriate database fields upon insertion into the database.

Vector Graphics: Illustrator (native), Freehand and Illustrator editable EPS, and PDF files can be inserted into the MediaServer system and exported in their original format. During insertion, a high quality thumbnail and preview is generated and all text elements contained within the graphic are extracted and indexed for searching by the Autonomy search engine.

Text: Supported text formats include AP NewsDesk, ANPA Wire Service (Bulletin 1312), Lexis Nexis Feed, coded text from SII, Atex and Harris editorial systems, NewsView Solutions FolioViews, NewsBank SAVE and DataTimes archive formats, plus standard text formats such as XML, Delimited Text (single record per file), Delimited Text (multiple records per file), and Bounded Text format. The Delimited Text format is a standard used by many applications, such as FileMaker, Microsoft Access, Excel, and just about any application that exports tabular data. Bounded Text format is a variation of delimited text in which each field is bounded by a unique start and end string.

Acrobat PDF: Any file, Graphics, Adverts, Pages, etc., saved in Acrobat PDF format.

Postscript: Raw postscript files, such as those created by the “Print To File” feature in printer drivers, are inserted with high quality thumbnails and previews. All text within the document represented by the postscript data is indexed so as to become fully searchable.

Multimedia: SCC MediaFactory supports the automatic extraction of thumbnail previews from QuickTime Movies during insertion into the MediaServer database. All other multimedia formats not containing embedded preview information such as WAV, MPEG, MP3, and WMV, are inserted with a generic thumbnail and searchable by Filename and any manually entered information.

Other file types: Any file can be inserted into MediaServer as a “foreign” file. Attributes can then be edited and searched and the file can be exported in its original form.

MediaServer exports files in their original format. File conversion to other formats can be accomplished automatically using MediaFactory image and text conversion channels, MediaGrid’s Launch Photoshop Action feature, or the MediaFactory AGS channel.

The SCC MediaServer system supports exporting high-resolution files from any network mountable volume in a single streamlined operation. The client is not required to mount any server volumes to export files from the database.

SCC MediaFactory supports the automatic processing and insertion of JPEG photos, and Freehand and PDF vector graphics from the AP server into the MediaServer database. Thumbnails, previews, and embedded IPTC Caption information is extracted and automatically mapped to fields within the MediaServer database.

The MediaServer system allows database records to be organized into “Projects”. A project is a list of objects collected by a user from one or more libraries. Users can create new projects, drag (add) objects into projects, show objects within a project, search objects within a project using the Autonomy search engine, remove objects from projects, and delete projects. Objects in a library can belong to one or more projects.

Projects are dynamic, just like libraries. That is, when an object is added to a project, the new object automatically appears in your search window if it meets the current search criteria and if that window is searching the project dynamically. Similarly, if objects are removed from a project they are removed from the view windows as well.

Objects in a project are not actually moved from the library, but are simply referenced by the project. In effect, when searching within a project, you are actually searching the libraries but the search results are additionally constrained to being contained in the project.

Objects in a project can be optionally hidden from certain users and groups by enabling this option in the MediaServer User Manager.

The MediaServer system supports “Multi-Object Linking,” which allows a user to link groups of files within the database to each other. In both MediaGrid and Web browser clients, linked files are displayed with a special icon. When clicked, all linked records, along with their source files, are retrieved and displayed.

Multi-Object links are multi-tiered, which means, among other things, that if one object in a pair of linked objects is linked to another object, the outer objects are not necessarily linked to each other.

Multi-Object Linking can be configured to run server-side, automatically linking objects that match pre-determined criteria, either in real-time, or at certain times of the day or night; or client-side using MediaGrid to manually drag and drop items onto each other, either within the same search results window or across multiple search results windows.

Editing

Both MediaGrid and Web Browser clients can be used for editing and enhancing object attributes. Users must connect to MediaServer with a Power User license to edit record attributes. MediaGrid, when used as a folder watcher, can also be used to edit and enhance JPEG image attributes before the file is inserted into the database.

The SCC Dialog Designer is an application that is supplied as part of the system. It enables the administrator to design custom display and editing forms to use different types of data.

The MediaServer Web Browser Access Module supports user-editable forms in an XML format. Standard HTML editing programs such as MacroMedia DreamWeaver can be used to create and edit these forms.

Several editing forms for both MediaGrid and Web browser clients are supplied with a MediaServer installation. For example, an IPTC caption editor form based on the typical attribute structure of photos and graphics is supplied, while a Text editor form matching a typical attribute structure for text records is supplied.

All dialogs delivered with the system allow editing via a simple mouse and/or keyboard cursor placement as well as text manipulation features commonly available in commercial word processing software (text selection via highlighting, cut, copy, paste, clear, etc.)

The administrator can customize display and editing forms to enable as many or as few attribute fields as desired to appear on the screen at any one time, depending on the available screen space. In addition, pop-up lists can be defined that allow faster entry by selecting data from predefined value lists.

The MediaGrid client allows a user to perform the tasks described for the Web Browser client, plus many others including:

- Batch infuse text into the same field of multiple records, as well as replace or append to existing keywords and other lists of text.
- Create and use AutoFill commands to greatly reduce the amount of editing time and increase the accuracy of repetitive tasks such as word count calculation, lead paragraph extraction, changing record status, and adding pre-determined values to fields such as Byline, Title, Date and Time.
- Check out a record for editing in Microsoft Word via the MediaServer Microsoft Word Integration Module.
- Multi-Object Linking via drag-and-drop.
- Add and remove objects to and from projects.

SCC MediaServer handles record locking to prevent concurrency conflicts. While records are locked as they are edited, they remain available for read-only browsing to other users. MediaGrid and Web browser users that attempt to edit a record that is already being edited will receive visual feedback that the record is locked and the object is read-only.

Both MediaGrid and Web browser clients allow data to be entered into each attribute field either manually and/or by pop-up lists. The type of data being entered into each field can also be controlled through field validation. For example, text cannot be entered into a date or number field.

“Launch Action” is a powerful feature that links MediaGrid to Adobe Photoshop. Photoshop version 5.0 or later has a feature called Actions, which allows the creation of complex image processing macros. These macros can be used to resample, filter, color balance, convert to another image file format, or call any set of Photoshop functions in sequence. Actions are initially created and named in Photoshop and this name is referenced in MediaGrid during the creation of Launch Action commands. MediaGrid’s Launch Action setup dialog allows the selection of the final output format and destination. Once created, a Launch Action command can either be selected from a pull-down menu, or included as part of an Automation script, making it easy to convert selected images to GIF, TIFF, DCS, EPS, or any other Photoshop-supported format. In addition, Launch Action is integrated with SCC’s Plug-In Kit for Photoshop to allow automatic implementation of softcrop dimensions as the Photoshop Action is being processed.

Proofing

SCC MediaServer supports the proofing of text and previews without the need to open the files within their native applications.

The SCC Layout Designer application allows the creation of custom proof layouts in a variety of formats. Any number of user-defined attributes can be included on the proof together with the image preview, optional variable parameters, and static text values.

SCC MediaServer supports printing single records or multiple records in batches, either client-side via the execution of a single command or server-side using the MediaServer Delivery Module with a single command or automatically with a persistent command. Any number of persistent commands can be configured to trigger automatically based on time and/or matching criteria within records.

Performance Issues

Search requests are performed as a server function and do not require continual network traffic between the server and the client. Only data necessary for display on the client screen is transferred once the search results are known. In addition, MediaGrid has a “smart” caching mechanism that minimizes data transfer between the server and the client. For example, if the user initiates a search, the search itself is performed on the server and the search results are passed back to the client as a list of IDs that uniquely identify the query results set. The client then requests the data needed to fill the first screen. If the user elects to scroll down, the client software will request the data necessary to fill the second screen. If the user then scrolls back up, the client uses its cached data. As the user scrolls through a large result set the client will automatically discard thumbnail data from offscreen cells as available memory becomes low.

MediaServer data is protected by transaction-based processing mechanisms available within SQL Server. Similarly, Autonomy search indexes are protected from corruption by transaction-based processing mechanisms inside the Autonomy search engine.

Support Issues

MediaServer sites choose between a Standard Support Package, which offers 9am-6pm Eastern Time telephone access to a qualified support Technician on business days (excludes seven holidays) or an Extended Support Package, which includes telephone access to a support technician 24x7x365, with one hour response time outside of normal business hours.

After system training of customer personnel has taken place and provided recommended procedures (including backups) are followed then technical support can be carried out remotely. On-site visits by SCC personnel are available and chargeable at the normal daily rate plus travel and accommodation expenses.

SCC requires that Symantec pcAnywhere remote access software is installed on each SCC server to allow access to the system by SCC technicians when necessary. Network access to the system should be made available via a secure VPN Internet account that is enabled and disabled by the customer as required. When remote access is prohibited by site policy, technical support is performed by voice conversation.

SCC recommends a superuser training and installation schedule based on a minimum of 10 days. Additional training can be scheduled as necessary. Extended installation periods may be necessary for larger installations with more complex workflow environments, or where SCC is required to perform end user training.

SCC MediaGrid Search Forms

There are two special search forms provided for MediaServer database searches and two special forms for searching files in folders. For MediaServer database searching there are the **Autonomy Search Form** and the **Production Search Form**. For file searching there are the **5-Level Search Form** and the **Any Text Search Form**.

Autonomy Search Form

The Autonomy Search Form, shown below, provides a powerful set of user controls for specifying a search expression. These controls include:

- a Search History Pulldown List containing the last 25 searches performed,
- a control to limit the hits per library,
- a 'Hits Are Most' control to choose whether hits are limited by relevance, insertion time, or content date,
- a Query Expression Field for entering any Autonomy Search Expression,
- a control to select the Autonomy Query Parser - normal expressions or natural language,
- a List of Libraries showing all the libraries currently open in the watch space and all libraries that should be included in the search,
- a Date Range control set for limiting searches to a range of dates,
- a 2-level Field Search,
- a Window Title Field, and
- a control to turn Dynamic Searches on and off.

The controls in the Autonomy Search Form can be divided into two types: controls to specify a query expression and supporting controls. Supporting controls include the Search History, Hit Limit, Library List, Window Title, and Dynamic Search controls. These controls are not used to describe a query expression but do, however, provide important features related to performing a search.

The **Search History Pulldown List** stores the last 25 searches performed with this search form. Selecting a previous search from the list will automatically populate all the query expression controls with the proper values. The supporting controls are left unchanged, thereby allowing previous searches to be performed with a different hit limit, on a different set of libraries, and so on.

The **Hit Limit** control is a pulldown list containing a range of hit limits from 50 to 2000. This is the maximum number of hits per library that will be displayed in the search results window.

The **Hits Are Most** control is a pulldown list containing a list of criteria to determine the relevance of the search criteria based on how recently it was inserted into the archive or how recent the content is. Choose from Relevant, Recent Insertion, and Recent Content.

The **Library List** shows all the libraries that have been attached to the current watchspace. However, even if a library is attached to a watchspace it does not have to be included in a particular search. Libraries that are not included in the search are dimmed. To make a library the only one to be searched, double-click it in the list. This will dim all the other libraries. To selectively and accumulatively include or exclude libraries from the search, double-click the library while holding down the Control Key. Note that this user-interface behavior is exactly the same as the user-interface in the Object List.

The **Window Title** field is available in all MediaGrid Search Forms and contains the base name of the window that will display the search results.

The **Dynamic Search** checkbox is also available in all MediaGrid Search Forms and designates whether or not the search results will be updated continually. When Dynamic Search is checked MediaGrid will set up an “active search agent”. This search agent will check each changed object as well as each inserted and deleted object and cause the search results window to update as needed to ensure that the results reflect the latest changes in the libraries.

The remaining controls are designed for entry of the query expression itself. There are two separate sections in the dialog. The Autonomy Search Query field is the large field in the upper half of the dialog and the Date Range and Field Search “helper” controls are in the lower half of the dialog.

The **Autonomy Search Query** field is for entering valid Autonomy query expressions. These expressions can be very simple or very complex. For example, in the figure shown above, the phrase “presidential debates” will be matched against all the text in any text field within the specified libraries. The combination of words “presidential debates”, and all stemmed versions of these words must exist as a phrase (adjacent words) for a match to occur. Alternatively, the expression could be “presidential <and> debates” to match any object in the libraries that contain both words but not necessarily as a phrase. Query expressions can be very complex, specifying date ranges, field searches, thesaurus, sound-alike (soundex), wildcard, proximity, and similarity searches. In fact, anything that can be specified in the date range and field search “helper” controls can also be specified in the Autonomy Search Query field if the appropriate syntax is used. Advanced users may choose this technique while novice users will find that the “helper” controls make date range and field searches very easy to specify. A detailed description of the Autonomy Search Query expression syntax is provided in a separate document named [Autonomy Query Language Manual](#).

As mentioned above, any search query can be entered into the Autonomy Search Query field. However, certain “helper” controls are provided for commonly used searches such as **Date Range and Field Searches**.

Date fields that can be applied to a date range search include, but are not limited to, Content Date, Creation Date, Insertion Date, Last Publication Date, and Last Write Time (modification date). Note that only the date portion is applied, not the time if it exists. Date range searches are not limited to this list of date fields, but rather can use any date or date-time field that might be added as a custom library attribute. The date attribute is selected from the Date Range attribute pulldown menu. To its left are two additional pulldown controls for specifying the start and end dates of the date range. These provide selection from a pulldown list or direct entry into the field. For example, you can use the pulldown to select the special designators ‘Today’, ‘Tomorrow’, or ‘Yesterday’, or you can enter a specific date. The special ‘day designators’ are equivalent to their respective dates at the moment of searching. (‘Today’ is equivalent to the current date.) You can also add or subtract days from these designators, as in ‘Today - 90’ which is equivalent to the date 90 days prior to the current date. The special designators not only provide a convenient shorthand, but they are also useful in saved searches because they always have the same meaning relative to the current date independent upon when they are loaded -- that is, the range from ‘Today - 7’ to ‘Today’ will always mean the previous seven days. Leaving the **From** field empty defines an open range that includes all dates on or before the **To** date. Leaving the **To** field empty defines an open range that includes all dates on or after the **From** date.

Field searches can be performed from the Autonomy Search Query field by entering the appropriate Autonomy expression, such as ‘Clinton <in> caption’. For novice users the **Field Search** controls provide a clear and easy mechanism for specifying the same thing. The controls are organized as a two-level field search. The field name (attribute name) is selected from the pulldown list at the left. Depending upon the type of field (text or date) the **Relationship** pulldown will fill with appropriate choices, such as ‘Contains’, ‘Does not contain’, ‘On or After’, and so on. The value to match is entered into the field at the right. Wildcard, <soundex>, and <thesaurus> operators can be inserted into the value when the attribute is a text type. The logical binding operator between the two levels of search is selected from the (none, AND, OR) control just beneath the first level of controls and the second level of field search can then be specified. Advanced users can enter additional levels of field search using the **Autonomy Search Query** field with the appropriate expression, such as ‘clinton <in> caption <and> clinton <in> headline <and> nws <in> category’.

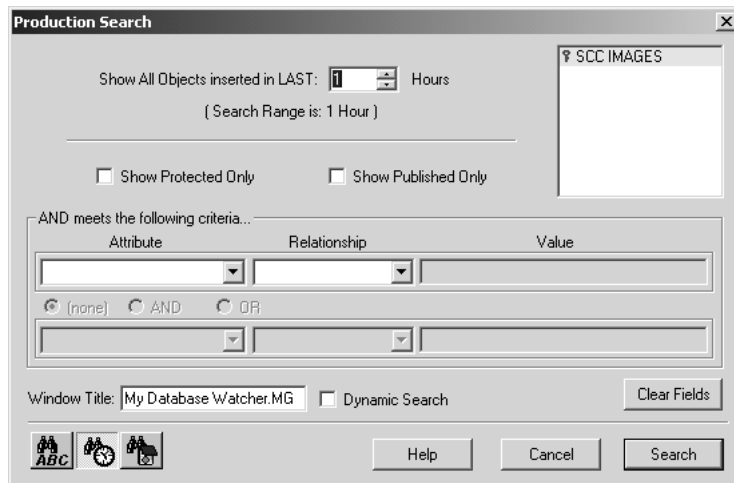
The **Clear All** button clears the “helper” controls and ensures that they will not contribute to the overall search expression that is generated. If the “helper” fields are not empty then they will contribute to the overall search expression with a logical <and> operator, meaning that they must be satisfied *in addition to* the expression entered into the **Autonomy Search Query** field, if any. To form a simple range search with the Date Range controls, for example, the **Autonomy Search Query** field must be cleared. Otherwise, the search will contain components from both areas and both components must be satisfied in order to match the overall search expression.

.....

Production Search Form

The Production Search Form is designed for workflows that require viewing the last several hours of files that have been inserted into the library or libraries. The number of hours is adjustable from 1 to 672 hours (4 weeks). This search is performed without any use of the Autonomy Search Engine -- the insertion time within the SQL Server database is used to determine which files meet the search criteria.

The Production Search Form is relatively simple. It contains an adjustment for the number of hours of files that should be displayed, the List of Libraries that will be searched, a constraint to show Protected files only, a constraint to show Published files only, and the Window Title and Dynamic Search controls that are common to all search forms.



The production search also includes field search controls that allow the search to be further constrained by up to two library attributes. Note that in order to be selectable from the attribute control, the system administrator must have previously configured these attributes to be “Production Searchable” within the MediaServer Administration Utility.

.....

Assignment Search Form

The Assignment Search Form is designed for search and retrieval of assignment objects and is available when the optional MediaServer Assignment Module is licensed. Only assignment libraries appear in the Assignment Library pulldown. Within this selected library search queries can be issued by Photographer, by Status, by date range, by Autonomy query statement, or any combination.

The screenshot shows a web-based search form titled "Assignment Search". The form contains the following elements:

- Assignment Library:** A dropdown menu with "ASSIGNMENT LIBRARY" selected.
- Limit hits to:** A dropdown menu with "50 Hits" selected.
- User:** A section with a checked checkbox "Assignments by Photographer" and a text input field containing "David Holloway".
- Status:** A section with an unchecked checkbox "Assignments by Status" and an empty text input field.
- Date Range:** A section with a dropdown menu "Insertion Time", and "From" and "To" dropdown menus with "Yesterday" and "Today" selected respectively.
- Query:** A large text input field containing "World Series".
- Window Title:** A text input field containing "My Assignments".
- Dynamic Search:** A checked checkbox.
- Buttons:** "Cancel", "Clear", and "Search" buttons.
- Accessibility:** Three icons at the bottom left: a keyboard (ABC), a wheelchair, and a person with a cane.

5-Level Search Form

The 5-Level search form is provided for file searching in a Folder Watchspace. It is not provided as a MediaServer database search form. The 5-Level Search form enables searching through files in one or more watch folders for specific values within embedded metadata fields. Using the multi-level nature of the form, up to 5 different fields can be included in the search criteria.

The screenshot shows a dialog box titled "Folder Search Forms" with a close button in the top right corner. The dialog is divided into several sections:

- History:** A dropdown menu showing "Caption CONTAINS sports OR City CONTAINS atlanta OR City CONTAINS chicago".
- Level 1:** A row with three columns: "Attribute" (Caption), "Relationship" (Contains), and "Value" (world series). Below this row are radio buttons for "(none)", "AND", and "OR", with "OR" selected.
- Level 2:** A row with three columns: "Attribute" (City), "Relationship" (Contains), and "Value" (atlanta). Below this row are radio buttons for "(none)", "AND", and "OR", with "AND" selected.
- Level 3:** A row with three columns: "Attribute" (Category), "Relationship" (Is Equal To), and "Value" (S). Below this row are radio buttons for "(none)", "AND", and "OR", with "(none)" selected.
- Level 4:** A row with three empty columns. Below this row are radio buttons for "(none)", "AND", and "OR", with "(none)" selected.
- Level 5:** A row with three empty columns. Below this row are radio buttons for "(none)", "AND", and "OR", with "(none)" selected.
- Options:** A checkbox for "Show All" (unchecked), a "Window Title:" field containing "World Series", a checked checkbox for "Dynamic Search", and a checked checkbox for "Match Partial Words".
- Buttons:** "Search", "Cancel", and "Help" buttons at the bottom right.

This search form contains a **Search History Pulldown List** that saves the last 25 searches performed with this search form. Selecting from this list will automatically populate each of the appropriate controls to duplicate the selected search criteria.

In each of the five levels, the attribute name is selected from the pulldown list at the left. Depending upon the type of field (text or date) the **Relationship** pulldown will fill with appropriate choices, such as 'Contains', 'Begins With', 'Is Greater Than', and so on. The value to match is entered into the field at the right.

The **Window Title** field is available in all MediaGrid Search Forms and contains the base name of the window that will display the search results.

The **Dynamic Search** checkbox is also available in all MediaGrid Search Forms and designates whether or not the search results will be updated continually. When Dynamic Search is checked MediaGrid will set up an "active search agent". This search agent will check each changed object as well as each new and deleted object and cause the search results window to update as needed to ensure that the results reflect the latest changes in the watch folders.

The **Show All** checkbox, when checked, disables the search criteria and forces the search results to include every file in the watch folders.

'Any Text' Search Form

The Any Text search form is provided for file searching in a Folder Watchspace. It is not provided as a MediaServer database search form. The Any Text Search form enables searching through the files within the watch folders for specific text within any text field.

The screenshot shows a dialog box titled "Folder Search Forms" with a close button (X) in the top right corner. At the top, there are two tabs: "5 Level" and "Any Text", with "Any Text" being the active tab. Below the tabs is a "History:" dropdown menu containing the text "world series". In the center of the dialog is a text input field with the label "Search for phrase in any text attribute:" and the text "grand slam" entered. At the bottom left, there is a checkbox labeled "Show All" which is unchecked. To its right is a "Window Title:" label followed by a text input field containing "World Series". Further right are two checked checkboxes: "Dynamic Search" and "Match Partial Words". At the bottom right of the dialog are three buttons: "Search", "Cancel", and "Help".

This search form contains a **Search History Pulldown List** that saves the last 25 searches performed with this search form. Selecting from this list will automatically populate each of the appropriate controls to duplicate the selected search criteria.

This is the simplest of all the search forms, providing essentially only one field for searching. Any text entered into this field will be used to find files with any text field matching this text.

The **Window Title** field is available in all MediaGrid Search Forms and contains the base name of the window that will display the search results.

The **Dynamic Search** checkbox is also available in all MediaGrid Search Forms and designates whether or not the search results will be updated continually. When Dynamic Search is checked MediaGrid will set up an "active search agent". This search agent will check each changed object as well as each new and deleted object and cause the search results window to update as needed to ensure that the results reflect the latest changes in the watch folders.

The **Show All** checkbox, when checked, disables the search criteria and forces the search results to include every file in the watch folders.

Autonomy Query Language

(This Query Language description has been extracted from documentation provided by Autonomy, Inc, and modified appropriately for the SCC MediaServer application). The Search Technology has been licensed from Autonomy, Inc and incorporated into the SCC MediaServer product.

Elements of Query Expressions

*A query expression is any statement you enter into the **Search Query** field within the **Search Form** as criteria for performing a search. The words and operators you use in a query expression comprise its elements.*

Simple and Explicit Syntax

*When the **Normal** query parser is selected in the **Search Form** you can state a query in either **Simple** or **Explicit** syntax. The syntax you use determines whether the search words you enter will be stemmed, whether the words that are found will contribute to relevance-ranked scoring, and so on, as described in the following sections of this document.*

*If there is an absence of **Explicit Syntax** operators and modifiers, your query expression is in **Simple Syntax**. When you use simple syntax, the search engine implicitly interprets single words you enter as if they were preceded by the **MANY** modifier and the **STEM** operator (described below). By implicitly applying the **MANY** modifier, the search engine calculates each document's score based on the word density it finds; the denser the occurrence of a word in a document, the higher the document's score. Scores are displayed in **MediaGrid** within the **Object List** and optionally in a column of a multi-column view window, among other locations.*

As a result, the search engine relevance-ranks documents according to word density as it searches for the word you specify, as well as words that have the same stem. For example, "films", "filmed", and "filming" are stemmed variations of the word "film". To search for documents containing the word "film" and its stemmed words, you can enter the word "film" using the simple syntax:

film

*When documents are relevance-ranked, they can be listed in order based on their relevance to your search criteria by selecting the sort order based on **Score**.*

*When you enclose individual words in double quotation marks, the search engine interprets those words literally. For example, by entering the word "film" explicitly in double-quotation marks, the words "films", "filmed", and "filming" will **NOT** be considered in the search. To select documents containing the word "film" without searching for its stemmed words, you can enter the word "film" using the explicit syntax:*

"film"

The following example retrieves documents that contain both the literal phrase “pharmaceutical companies” and the literal word “stock”:

```
AND ("pharmaceutical companies" , "stock")
```

or you can use the syntax

```
"pharmaceutical companies" AND "stock"
```

The operator AND does not have to be upper case nor does it have to be surrounded by angle brackets, as do most operators, because it is automatically interpreted as an operator.

The following explicit syntax example retrieves documents containing the phrase “pharmaceutical companies”

```
<PHRASE> (pharmaceutical companies)
```

However, the following simple syntax example accomplishes the same result:

```
pharmaceutical companies
```

As illustrated above, explicit syntax is often unnecessary. For the advanced user, however, explicit syntax operators and modifiers provide a very powerful set of search features. These features are described with their explicit syntax in the following sections of this document and, when appropriate, comparative examples of simple syntax are provided.

Operator Summary

An operator represents logic to be applied to a search element. This logic defines the qualifications a document must meet to be retrieved. Many operators are available, and they can be classified by their type, as follows:

- Evidence operators
- Proximity operators
- Concept operators

Modifiers extend the logic applied by operators and are used in combination with operators. Modifiers are discussed in detail in a later section (See “Modifier Reference” on page 44.)

Evidence Operators

Evidence operators can be used to specify either a basic word search or an intelligent word search. A basic word search finds documents that contain only the word or words specified in the query. An intelligent word search expands the query terms to create an expanded word list so that the search returns documents that contain variations of the query terms. For example, the THESAURUS operator selects documents containing the word specified, as well as its synonyms.

Documents retrieved using evidence operators are not relevance-ranked unless you use the MANY modifier. Refer to “MANY Modifier” for information (See “MANY Modifier” on page 44.). The following table describes each evidence operator.

Operator Name	Description
WORD	Performs a basic word search, selecting documents that include one or more instances of the specific word you enter.
STEM	Expands the search to include the word you enter and its variations.
THESAURUS	Expands the search to include the word you enter and its synonyms.
TYPO/ <i>n</i>	Expands the search to include the word you enter plus words that have a similar spelling.
WILDCARD	Matches wildcard characters included in search strings. Certain characters automatically indicate a wildcard specification.
SOUNDEX	Expands the search to include the word you enter and one or more words that “sound like,” or whose letter pattern is similar to, the word specified. Collections do not have sound-alike indexes by default; to use this feature sound-alike indexes must be built.

The STEM operator is implicitly assumed when the query expression is in simple syntax -- that is, in the absence of one of the operators in the table above. As seen, surrounding a word with double-quotation marks is equivalent to using the WORD operator, which avoids the stemming function, however it also implicitly adds the MANY operator to get relevance-ranking.

Using STEM, THESAURUS, WILDCARD, or SOUNDEX operators cause the word to be expanded in to a list of words. In the case of STEM, the words in the expanded list are the stemmed words, as in film, films, filmed, and filming. For THESAURUS, the words in the list are synonyms; for WILDCARD they are all words expanded appropriately around the wildcard characters; and for SOUNDEX the list consists of words that sound alike. This list of words is then used by the search engine to generate a new search expression such that the search is performed over the entire list. Further discussion and examples of the Evidence Operator are provided later in this document (See “Operator Reference” on page 37.).

Proximity Operators

Proximity operators specify the relative location of specific words in the document; that is, specified words must be in the same phrase, paragraph, or sentence for a document to be retrieved. In the case of the NEAR and NEAR/N operators, retrieved documents are relevance-ranked based on the proximity of the specified words. When proximity operators are nested, the ones with the broadest scope should be used first; that is, phrases or individual words can appear within SENTENCE or PARAGRAPH operators, and SENTENCE operators can appear within PARAGRAPH operators. The following table describes each proximity operator.

Operator Name	Description
IN	Selects documents that contain specified values in one or more MediaServer Library attributes (fields).
PHRASE	Selects documents that include a phrase you specify. A phrase is a grouping of two or more words that occur in a specific order.
SENTENCE	Selects documents that include all of the words you specify within the same sentence.
PARAGRAPH	Selects documents that include all of the search elements you specify within the same paragraph.
NEAR	Selects documents containing specified search terms, where the closer the search terms are within a document, the higher the document's score.
NEAR/n	Selects documents containing two or more search terms within N number of words of each other, where <i>n</i> is an integer up to 1000. The closer the search terms are within a document, the higher the document's score.

Additional discussion and examples are provided later in this document (See “Operator Reference” on page 37.).

Concept Operators

Concept operators combine the meaning of search elements to identify a concept in a document. Documents retrieved using concept operators are relevance ranked. The following table describes each concept operator.

Operator Name	Description
AND	Selects documents that contain all of the search elements you specify.
OR	Selects documents that show evidence of at least one of the search elements you specify.
ACC RUE	Selects documents that include at least one of the search elements you specify. The more search elements that are present, the higher the score will be.

Additional discussion and examples are provided later in this document (See “Operator Reference” on page 37.).

Operator Reference

This section describes each operator in detail. Where appropriate, each description includes an example of simple syntax and explicit syntax. Operators are listed alphabetically.

ACCRUE Operator

Selects documents that include at least one of the search elements you specify. Valid search elements are two or more words or phrases. Retrieved documents are relevance-ranked.

The ACCRUE operator scores retrieved documents according to the presence of each search element in the document using “the more, the better” approach: the more search elements found in the document, the better the document’s score. Following are examples of search syntax.

To select documents containing stemmed variations of the words “computers” and “laptops,” you can enter any of the following:

```
computers <ACCRUE> laptops
computers, laptops
<ACCRUE> (computers, laptops)
```

AND Operator

Selects documents that contain all of the search elements you specify. Documents retrieved using the AND operator are relevance-ranked. Following are examples of search syntax.

To select documents which contain stemmed variations of the phrase “pharmaceutical companies” and stemmed variations of the word “stock,” you can enter the following:

```
pharmaceutical companies AND stock
```

Only those documents that contain both search elements, or stemmed variations of them (for example, “pharmaceutical company,” “stocks,” etc.), are retrieved and ranked according to their scores.

> (GREATER THAN) Operator

Selects documents whose document field values are greater than the search string you specify. This operator is used with numeric and date attributes (fields). For example, assume a document field named DATE has been defined. To select only those documents dated after October 24, 1992, you can enter the following:

```
DATE > 10-24-92
```

>= (GREATER THAN OR EQUAL TO) Operator

Selects documents whose document field values are greater than or equal to the search string you specify. This operator is used with numeric and date attributes (fields). For example, assume a document field named DATE has been defined. To select only those documents dated on or after October 24, 1992, you can enter the following:

```
DATE >= 10-24-92
```

< (LESS THAN) Operator

Selects documents whose document field values are less than the search string you specify. This operator is used with numeric and date attributes (fields). For example, assume a document field named DATE has been defined. To select only those documents dated before February 14, 1991, you can enter the following:

```
DATE < 02-14-91
```

<= (LESS THAN OR EQUAL TO) Operator

Selects documents whose document field values are less than or equal to the search string you specify. This operator is used with numeric and date attributes (fields). For example, assume a document field named DATE has been defined. To select only those documents dated prior to and including February 14, 1991, you can enter the following:

```
DATE <= 02-14-91
```

IN Operator

Selects documents that contain specified values in one or more attributes (fields). The IN operator is inserted between the attribute name and the value to be matched. Spaces in attribute names, such as in 'body text', should be replaced by a '-' character, as in 'body-text', whenever entering an attribute name in the Search Query field (in the Search Form).

The following query expression searches the attributes named "body-text" for the word "safety."

```
"safety" <IN> body-text
```

To search with multiple words or phrases enclose them in parentheses. The following query expression searches the attribute named "body-text" for the word "safety" and stemmed variations of the word "warning."

```
("safety", warning) <IN> body-text
```

To search multiple attributes, separate them with commas and enclose them in parentheses. The following query expression searches both the "body-text" attribute and the "title" attribute for the word "safety" and stemmed variations of the word "warning."

```
("safety", warning) <IN> (body-text, title)
```

You must enclose query expressions containing commas in parentheses. The following example searches the “body-text” attribute for the word “safety” and stemmed variations of the phrase “federal regulation.”

```
("safety", federal regulation) <IN> body-text
```

The following query expression searches both the “body-text” attribute and the “title” attribute for the word “safety” and stemmed variations of the phrase “federal regulation.”

```
("safety", federal regulation) <IN> (body-text, title)
```

NEAR Operator

Selects documents containing specified search terms within close proximity to each other. Document scores are calculated based on the relative number of words between search terms. For example, if the search expression includes two words, and those words occur next to each other in a document (so that the region size is two words long), then the score assigned to that document is 100. Thus, the document with the smallest possible region containing all search terms always receives the highest score. Documents whose search terms are not within 1000 words of each other are not selected, since the search terms are probably too far apart to be meaningful within the context of the document.

The NEAR operator is similar to the other proximity operators in the sense that the search words you enter must be found within close proximity of one another. However, unlike other proximity operators, the NEAR operator calculates relative proximity and assigns scores based on its calculations.

NEAR/n Operator

Selects documents containing two or more words within n number of words of each other, where n is an integer. Document scores are calculated based on the relative distance of the specified words when they are separated by n words or less.

For example, if the search expression NEAR/5 is used to find two words within five words of each other, a document that has specified words within three words of each other is scored higher than a document that has specified words within five words of each other.

The n variable can be an integer between 1 and 1,024, where NEAR/1 searches for two words that are next to each other. If n is 1,000 or above, you must specify its value without commas, as in NEAR/1000. You can specify multiple search terms using multiple instances of NEAR/n, as long as the value of n is the same.

For example, to retrieve relevance-ranked documents that contain stemmed variations of the words “commute,” “bicycle,” “train,” and “bus” within 10 words of each other, you can enter the following:

```
commute <NEAR/10> bicycle <NEAR/10> train <NEAR/10> bus
```

You can use the NEAR/n operator with the ORDER modifier to perform ordered proximity searches. For more information about the ORDER modifier (See “ORDER Modifier” on page 45.).

OR Operator

Selects documents that show evidence of at least one of your search elements. Documents selected using the OR operator are relevance-ranked.

To select documents that contain stemmed variations of the word “election” or the phrases “national elections” or “senatorial race”, you can enter the following:

```
election OR national elections OR senatorial race
```

Only those documents that contain at least one of the search elements, or a stemmed variation of at least one of them, are retrieved and ranked according to their scores.

Paragraph Operator

Selects documents that include all of the search elements you specify within a paragraph. Valid search elements are two or more words or phrases. You can specify search elements in a sequential or a random order. Documents are retrieved as long as search elements appear in the same paragraph.

To retrieve relevance-ranked documents that contain stemmed variations of the word “drug” and the phrase “cancer treating” in the same paragraph, you can enter the following:

```
drug <PARAGRAPH> cancer treating
```

To search for three or more words or phrases, you must use the PARAGRAPH operator between each word or phrase.

You can use the PARAGRAPH operator with the ORDER modifier to perform ordered proximity searches. For more information about the ORDER modifier (See “ORDER Modifier” on page 45.).

Important: Use of the PARAGRAPH operator is valid only on MediaServer libraries that have Paragraph parsing enabled. Libraries, such as those containing PDFs, might have Paragraph parsing disabled by the System Administrator. This is because PDF text extraction technology provided by Adobe does not distinguish between paragraphs, so the entire text content of the PDF is extracted as a single paragraph.

PHRASE Operator

Selects documents that include a phrase you specify. A phrase is a grouping of two or more words that occur next to each other in a specific order.

By default, two or more words separated by a space are considered to be a phrase in simple syntax. In addition, two or more words enclosed in double quotes are considered to be a phrase. To retrieve relevance-ranked documents that contain the phrase “mission oak,” you can enter any of the following:

```
mission oak  
"mission oak"  
mission <PHRASE> oak  
<PHRASE> (mission, oak)
```

SENTENCE Operator

Selects documents that include all of the words you specify within a sentence. You can specify search elements in a sequential or a random order. Documents are retrieved as long as search elements appear in the same sentence.

To retrieve relevance-ranked documents that contain stemmed variations of the words “American,” and “innovation” within the same sentence, you can enter the following:

```
american <SENTENCE> innovation  
<SENTENCE> (american, innovation)
```

You can use the SENTENCE operator with the ORDER modifier to perform ordered proximity searches. For more information about the ORDER modifier (See “ORDER Modifier” on page 45.).

Important: Use of the SENTENCE operator is valid only on MediaServer libraries that have Paragraph/Sentence parsing enabled. (See note in PARAGRAPH Operator description).

SOUNDEX Operator

Selects documents that include one or more words that “sound alike,” or whose letter pattern is similar to, the word specified. Words have to start with the same letter as the word you specify to be selected.

For example, to retrieve documents containing a word that is close in structure to the word “sale,” you can enter the following:

```
<SOUNDEX> sale
```

The documents retrieved will include words such as “sale,” “sell,” “seal,” “shell,” “soul,” and “scale.” Documents are not relevance-ranked unless the MANY modifier is used, as in:

```
<MANY><SOUNDEX> sale
```

STEM Operator

Selects documents that include one or more variations of the search word you specify. For example, to retrieve documents containing a variation of the word “film,” you can enter the following:

```
<STEM> film
```

The documents retrieved will include words such as “films,” “filmed,” and “filming.” Documents are not relevance-ranked unless the MANY modifier is used, as in:

```
<MANY><STEM> film
```

In simple syntax, both the MANY and STEM operators are implied for each word.

THESAURUS Operator

Selects documents that contain one or more synonyms of the word you specify. For example, to retrieve documents containing synonyms of the word “altitude” you can enter the following:

```
<THESAURUS> altitude
```

The documents retrieved will include words such as “height” or “elevation.” Documents are not relevance-ranked unless the MANY modifier is used, as in:

```
<MANY><THESAURUS> altitude
```

TYPO/n Operator

Selects documents that contain the word you specify plus words that are similar to the query term. The TYPO/n operator performs “approximate pattern matching” to identify similar words.

The optional n variable in the operator name expresses the maximum number of spelling errors between the query term and the matched term, where an error is defined as a character deletion, insertion, or transposition. If n is not specified, a value of 2 is used.

WILDCARD Operator

Selects documents that contain matches to a wildcard character string. The WILDCARD operator lets you define a wildcard string, which can be used to locate related word matches in documents. A wildcard string consists of special characters. For example, to retrieve documents that contain words such as, “pharmaceutical,” “pharmacology,” and “pharmacodynamics,” you can enter the following:

```
pharmac*
```

Documents are not relevance-ranked unless the MANY modifier is used, as in:

```
<MANY> pharmac*
```

The wildcard characters “*” and “?” automatically enable wildcard searching. To use other constructs, use the WILDCARD operator explicitly with any of the characters below.

Character	Function
?	Specifies one of any alphanumeric character, as in ?an, which locates “ran,” “pan,” “can,” and “ban.” Note that it is not necessary to specify the WILDCARD operator when you use the question mark. The question mark is ignored in a set ([]) or in an alternative pattern ({ }).
*	Specifies zero or more of any alphanumeric character, as in corp*, which locates “corporate,” “corporation,” “corporal,” and “corpulent.” Note that it is not necessary to specify the WILDCARD operator when you use the asterisk; you should not use the asterisk to specify the first character of a wildcard string . The asterisk is ignored in a set ([]) or in an alternative pattern ({ }).

Character	Function
[]	Specifies one of any character in a set, as in <WILDCARD> 'c[auo]t', which locates "cat," "cut," and "cot." You must enclose the word that includes a set in backquotes ('), and there can be no spaces in a set.
{ }	Specifies one of each pattern separated by a comma, as in <WILDCARD> 'bank{s,er,ing}', which locates "banks," "banker," and "banking." You must enclose the word which includes a pattern in backquotes ('), and there can be no spaces in a set.
^	Specifies one of any character not in the set, as in <WILDCARD> 'st[^oa]ck', which excludes "stock," and "stack" but locates "stick," and "stuck." Note that the caret (^) must be the first character after the left bracket ([) that introduces a set.
-	Specifies a range of characters in a set, as in <WILDCARD> 'c[a-r]t', which locates every three-letter word from "cat" to "crt."

Searching for Wildcard Characters as Literals

Provided the *style.lex* file is set up for the collections to be searched, you can search for a word containing a wildcard character such as "/" or "*" by preceding the wildcard character with a backslash. For example, if you enter the following search string:

```
abc\*d
```

the engine finds five-character words matching the "abc*d" string.

When you want to match a literal backslash, you must enter two backslashes.

Searching for Special Characters as Literals

The following nonalphanumeric characters perform special, internal search engine functions, and by default are not treated as literals in a wildcard string:

- comma ,
- left and right parentheses ()
- double quotation mark "
- backslash \
- at sign @
- left curly brace {
- left bracket [
- less than sign <
- backquote `

To interpret special characters as literals, you must surround the whole wildcard string in backquotes (`). For example, to search for the wildcard string "a{b", you surround the string with backquotes, as follows:

```
<WILDCARD> `a{b`
```

To search for a wildcard string that includes the literal backquote character (`), you must use two backquotes together and surround the whole wildcard string in backquotes (`), as follows:

```
<WILDCARD> ``n``t`
```

WORD Operator

Selects documents that include one or more instances of a word you specify. For example, to search for documents that contain the word “rhetoric,” without also considering the words “rhetorical” and “rhetorician,” you can enter the following:

```
<WORD> rhetoric
```

Documents are not relevance-ranked unless the MANY modifier is used, as in:

```
<MANY><WORD> rhetoric
```

Modifier Reference

Modifiers are used in conjunction with operators. When specified, a modifier changes the standard behavior of an operator in some way. For example, you can use the CASE modifier with an operator to specify that the case of the search word you enter be considered a search element as well. Modifiers include CASE, MANY, NOT, and ORDER, each of which is described below.

CASE Modifier

Use the CASE modifier with the WORD or WILDCARD operator to perform a case-sensitive search, based on the case of the word or phrase specified.

To use the CASE modifier, simply enter the search word or phrase as you wish it to appear in retrieved documents --- in all uppercase letters, in mixed uppercase and lowercase letters, or in all lowercase letters.

For example, to retrieve documents that contain the word “Apple” in mixed uppercase and lowercase letters, you can enter the following:

```
<CASE> <WORD> Apple
```

Only those documents that contain the word “Apple” will be selected. Occurrences of “apple,” “apples,” or “APPLE” will not be selected.

When mixed uppercase and lowercase characters are included in a query, the search engine finds case-sensitive matches.

MANY Modifier

Counts the density of words, stemmed variations, or phrases in a document, and produces a relevance-ranked score for retrieved documents. The more occurrences of a word, stem, or phrase proportional to the amount of document text, the higher the score of that document when retrieved. Because the MANY modifier considers density in proportion to document text, a longer document that contains more occurrences of a word can score lower than a shorter document that contains fewer occurrences. You can use the MANY modifier with these operators: WORD, WILDCARD, STEM, SOUNDEX, PHRASE, SENTENCE, PARAGRAPH.

For example, to select documents based on the density of stemmed variations of the word “apple,” you can enter the following:

```
<MANY> <STEM> apple
```

NOT Modifier

Use the *NOT* modifier with a word or phrase to exclude documents that show evidence of that word or phrase. For example, to select only documents that contain the words “cat” and “mouse” but not the word “dog,” you can enter the following:

```
cat <AND> mouse <AND> <NOT> dog
```

You can use the *NOT* modifier only with the operators *AND* and *OR*.

ORDER Modifier

Use the *ORDER* modifier to specify that search elements must occur in the same order in which they were specified in the query. If search values do not occur in the specified order in a document, the document is not selected. You can use the *ORDER* modifier with these operators: *PARAGRAPH*, *SENTENCE*, and *NEAR/n*.

Always place the *ORDER* modifier just before the operator. The following syntax examples show how you can use either simple syntax or explicit syntax to retrieve documents containing the word “president” followed by the word “washington” in the same sentence.

Simple syntax:

```
president <ORDER><PARAGRAPH> washington
```

Explicit syntax:

```
<ORDER><PARAGRAPH> ("president", "washington")
```

To search for documents containing the words “diver,” “kills,” “shark” in that order within 20 words of each other, use one of the following queries:

```
diver <ORDER><NEAR/20> kills <ORDER><NEAR/20> shark
```

```
<ORDER> <NEAR/20> (diver, kills, shark)
```

You can use the *NEAR/n* operator with the *ORDER* modifier to duplicate the behavior of the *PHRASE* operator. For example, to search for documents containing the phrase “world wide web,” you can use the following syntax:

```
world <ORDER><NEAR/1> wide <ORDER><NEAR/1> web
```

Score Operators

The score operators affect how the search engine calculates scores for retrieved documents. When a score operator is used, the search engine first calculates a separate score for each search element found in a document, and then performs a mathematical operation on the individual element scores to arrive at the final score for each document.

The *YESNO* operator has wide application, whereas the *PRODUCT*, *SUM*, and *COMPLEMENT* operators are intended for use mainly by application developers who want to generate queries programmatically.

YESNO Operator

Forces the score of an element to 100, if the element's score is nonzero. Examples help clarify this.

```
<YesNo> ("Chloe")
```

If the retrieval result of the search on "Chloe" was 75, with the YesNo operator, the result would be 100; if the retrieval result is 0, it remains 0.

This operator allows you to limit a search to only those documents matching a query, without the score of that query affecting the final scores of the documents. For example, to search among documents that contain "Chloe," with "Mead" as the determinant for ranking, you cannot simply specify the following:

```
"Chloe" <AND> "Mead"
```

because that would produce documents ranked with scores combined from both elements. The following would do what you want:

```
<YesNo> ("Chloe") <AND> "Mead"
```

If the retrieval result of the search on "Chloe" was 50 and that on "Mead" was 75, without the YesNo Operator, the combined result would be 50; with the operator, however, it is 75, because the score of AND is calculated to be the minimum score of all its search elements.

Product Operator

Calculates scores for documents matching a query by multiplying the scores for the query's search elements together. To arrive at a document's score, the search engine calculates a score for each search element and multiplies these scores together.

Following is an example of search syntax:

```
<PRODUCT> ("computers", "laptops")
```

If a search on "computers" generated a score of 50 and a search on "laptops" generated a score of 75, the preceding search would produce a score of 37.5.

SUM Operator

Calculates scores for documents matching a query by adding together, to a maximum of 100, the scores for the query's search elements. (Note: calculations are made using scores that are normalized to 1.0, so that a score of 75 is normalized to .75 in the calculation.) To arrive at a document's score, the search engine calculates a score for each search element and adds these scores together.

Following is an example query expression:

```
<SUM> ("computers", "laptops")
```

If a search on "computers" generated a score of 50 and a search on "laptops" generated a score of 20, the preceding search would produce a score of 70. If a search on "computers" generated a score of 50 and a search on "laptops" generated a score of 75, the preceding search would produce a score of 100 (the maximum).

COMPLEMENT Operator

Calculates scores for documents matching a query by taking the complement (subtracting from 100) the scores for the query's search elements. To arrive at a document's score, the search engine calculates a score for each search element and takes the complement of these scores.

Following is an example of search syntax:

```
<COMPLEMENT> ("computers")
```

The COMPLEMENT operator is a unary operator. It multiplies search elements as specified. The elements are combined, using the ACCRUE operator by default, to generate a single score which is then complemented. A sample query expression with two search elements is below:

```
<COMPLEMENT> ("computers", "laptops")
```

In the above example, the query is evaluated as the word "computers" accrued using the ACCRUE operator with the word "laptops." The COMPLEMENT operator is applied to the result.

Natural Language Operators

The natural language operators enable you to specify search criteria using natural language syntax. The search engine uses natural language analysis to translate the query text into query language expression for evaluating and scoring documents.

FREETEXT Operator

Interprets text using the free text query parser and scores documents using the resulting query expression. All retrieved documents are relevance-ranked.

This operator provides the functionality of the free text query parser, but allows you to combine free text queries with other search criteria using the full query language. For example:

```
<FREETEXT> ( "peace negotiations in the Middle East" ) <AND>  
(DATE > 01-01-96)
```

The quotation marks are required. If you want to include embedded quotes, they must be preceded with backslashes, as:

```
<FREETEXT> ( "\"Independence Day\"", ("\"The Arrival\"",  
science fiction" )
```

The FREETEXT operator can be combined with other operators in the same way as the ACCRUE operator.

LIKE Operator

Searches for other documents that are like the text passages you provide. The search engine analyzes the provided text to find the most important terms to use for the search. If multiple samples are provided, the search engine assumes all of the samples are about a single theme and selects important terms common across the samples. Retrieved documents are relevance-ranked.

The LIKE operator accepts a single operand, called the QBE (query-by-example) specification. The QBE specification can be either the literal text of the example to query on, or it can be a specification of one or more full documents and text passages to use as positive and negative examples.

In SCC MediaGrid, you can easily find similar documents through the LIKE search mechanism by invoking the 'Search For Similar By' menu command in the context menu for any cell.

Precedence Evaluation Rules

The language used for composing search statements has the following rules which determine the precedence for evaluation. The rules refer to Query Language operators.

Precedence Rules

A query expression is read using explicit precedence rules applying to the operators which are used. While a query expression is read from left to right, some operators carry more weight than others and this will affect the interpretation of the expression. For example, an AND operator takes precedence over an OR operator. For this reason, the following example is interpreted to mean: Look for documents that contain b and c, or documents that contain a.

a OR b AND c

To ensure that the OR operator is interpreted first, you can use parentheses as follows:

(a OR b) AND c

In general, the appropriate use of parentheses in query expressions, especially complex ones, will ensure that the query expression is interpreted as intended.

Parentheses in Expressions

Parentheses indicate the order the directions are to be carried out; information within parentheses is read first, then information outside parentheses is read next. Note that there must be at least one space between operators and words used in the expression. The following example means: Look for documents that contain a and b, or documents that contain c.

(a AND b) OR c

If there are nested parentheses, start with the innermost level. The following example means: Look for documents that contain b or c, as well as a, or that contain d.

(a AND (b OR c)) OR d

Prefix and Infix Notation

Words that use any operator except evidence operators (*SOUNDEX*, *STEM*, *THESAURUS*, *WILDCARD*, and *WORD*) can be defined in prefix notation or in infix notation.

Prefix notation is a format that specifies that the operator comes before the words or topics to be used with that operator. The following example means: Look for documents that contain a and b.

AND (a,b)

When prefix notation is used, precedence is explicit within the expression. The following example means: Look for documents that contain b and c first, then documents that contain a.

OR (a, AND (b,c))

Infix notation is a format that specifies that the operator is to be specified between each element within the expression. The following example means: Look for documents that contain a and b or documents that contain c.

a AND b OR c

When infix notation is used, precedence is implicit within the expression; for example, the AND operator takes precedence over the OR operator.

Commas in Expressions

If an expression includes two or more words within parentheses, as is used in prefix notation, a comma is required as a separator between each element enclosed within the parentheses.

The following example means: Look for documents that contain any combination of a and b together. Note that the OR operator is enclosed in angle brackets (< >), as described in “Angle Brackets for Operators” in this chapter.

<OR> (a, b)

Delimiters in Expressions

Angle brackets (< >), double quotation marks (“”), and backslashes (\) are used in expressions as described below.

Angle Brackets for Operators

Left and right angle brackets (< >) are reserved for designating operators and modifiers. **They are optional for AND, OR, and NOT, but required in all other cases.** Some examples in this guide appear with angle brackets, and some without. As the following simple syntax examples show, you can enter expressions either way:

future <AND> trends

future AND trends

Both expressions mean: Look for documents that contain the stemmed variations of the words “future” and “trends”.

Double Quotes for Reserved Words

To search for a word that is reserved as an operator (AND, OR, and NOT), enclose the word in double quotation marks. For example, to search for the phrase “black and white TV,” you can enter the following simple syntax:

```
black "and" white TV
```

Enclosing the word “and” in double quotation marks signifies that “and” should be considered as a word, not an operator.

Backslashes for Special Characters

To include a backslash (\) in a search, insert two backslashes for each backslash character. To search for “C:\bin\print,” you can enter the following simple syntax:

```
C:\\bin\\print
```